

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1.3
дисциплины «Алгоритмизация»

Выполнил:
Степанов Леонид Викторович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение
средств вычислительной
техники и автоматизирование
систем», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р.А., канд. техн. наук,
доцент, доцент кафедры
инфокоммуникаций

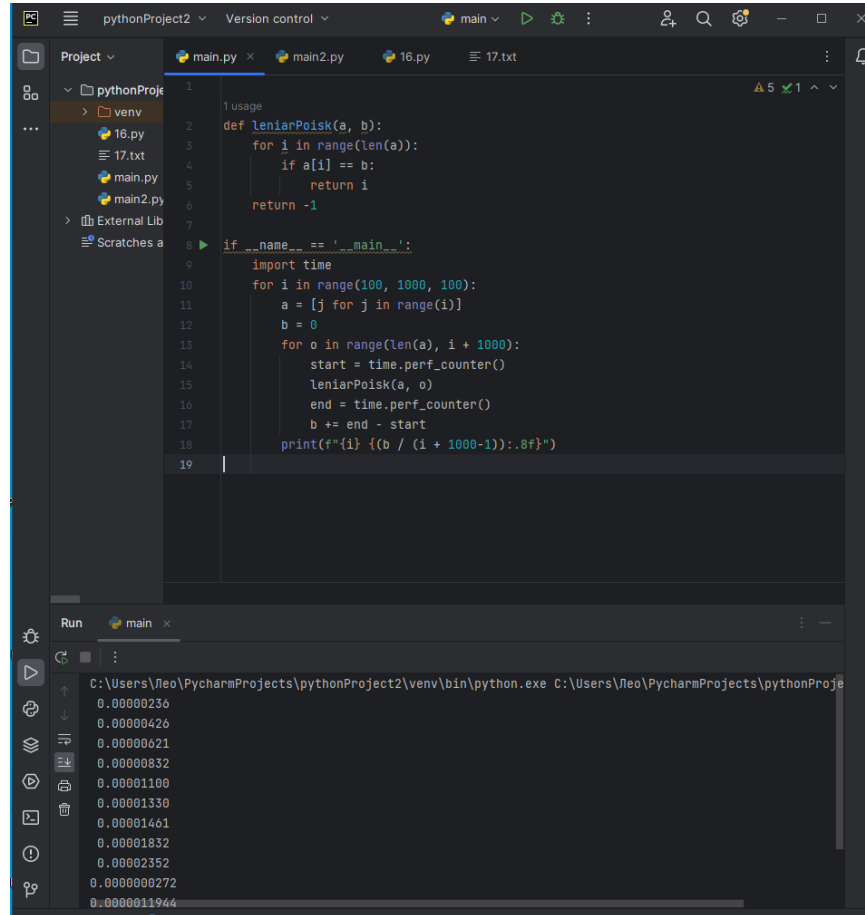
(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы:

1. Написал программу (linW.py), которая использует алгоритм линейного поиска, она рассчитывает какое время необходимо при худшем случае при разных значениях длины массива: от 100 до 1000 с шагом 100.



```
1 usage
2 def leniarPoisk(a, b):
3     for i in range(len(a)):
4         if a[i] == b:
5             return i
6     return -1
7
8 if __name__ == '__main__':
9     import time
10    for i in range(100, 1000, 100):
11        a = [j for j in range(i)]
12        b = 0
13        for o in range(len(a), i + 1000):
14            start = time.perf_counter()
15            leniarPoisk(a, o)
16            end = time.perf_counter()
17            b += end - start
18        print(f'i {i} {(b / (i + 1000 - 1)):.8f}")
19
```

Run console output:

```
C:\Users\Neo\PycharmProjects\pythonProject2\venv\bin\python.exe C:\Users\Neo\PycharmProjects\pythonProject2\linW.py
0.00000236
0.00000426
0.00000621
0.00000832
0.00001100
0.00001330
0.00001461
0.00001832
0.00002352
0.000000272
0.0000011944
```

Рисунок 1 – Результат выполнения программы linW.py

										summ
x	100	200	300	400	500	600	700	800	900	4500,0000000000
y	0,00000368	0,00000648	0,00000939	0,00001253	0,0000153	0,00001767	0,00001982	0,00002246	0,00002580	0,0001331330
x^2	10000,00	40000,00	90000,00	160000,00	250000,00	360000,00	490000,00	640000,00	810000,00	2850000,00
y*x	0,000368	0,001296	0,002817	0,005012	0,00765	0,010602	0,013874	0,017968	0,0232227	0,0828097000
y^2	0,0000000001354	0,0000000004199	0,0000000008817	0,0000000015700	0,0000000023409	0,0000000031223	0,0000000039283	0,0000000050445	0,0000000066579	0,00000000241010
Y	+ 0,0000011944	0,0000066344	0,0000093544	0,0000120744	0,0000147944	0,0000175144	0,0000202344	0,0000229544	0,0000256744	0,0001331496

Рисунок 2 – Таблица значений в excel

При помощи метода наименьших квадратов вывели систему уравнений: $2850000a + 4500b = 0,0828$ и $4500a + 9b = 0,000133$, решив которую мы нашли график функции: $y = 0,0000000272x + 0,0000011944$ и проверили при помощи метода корреляции правильность графика.

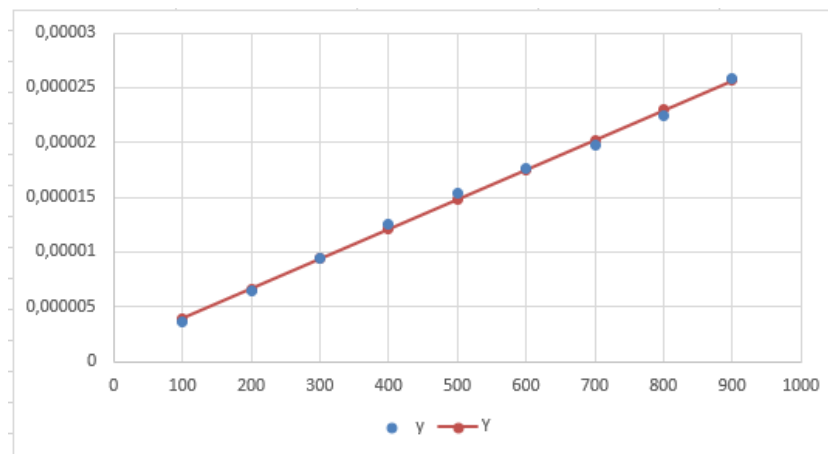


Рисунок 3 – График функции $y = 0,0000000272x + 0,0000011944$

Г-исх	0,998875882
R-кв	1

Рисунок 4 – Проверка метода наименьших квадратов

2. Написал программу (linS.py), которая использует алгоритм линейного поиска, она рассчитывает какое время необходимо при среднем случае при разных значениях длины массива: от 100 до 1000 с шагом 100

```

1 usage
2 def leniarPoisk(a, b):
3     for i in range(len(a)):
4         if a[i] == b:
5             return i
6     return -1
7
8 if __name__ == '__main__':
9     import time
10    for i in range(100, 1000, 100):
11        a = [j for j in range(i)]
12        b = 0
13        for o in range(len(a) - 1, 1, -1):
14            start = time.perf_counter()
15            leniarPoisk(a, o)
16            end = time.perf_counter()
17            b += end - start
18    print(f" {(b / (i - 3)):.8f}")
  
```

Run main

```

C:\Users\leo\PycharmProjects\pythonProject2\venv\bin\python.exe C:\Users\leo\PycharmProjects\pythonProject2\venv\bin\python.exe
0.00000239
0.00000418
0.00000605
0.00000818
0.00000984
0.00001210
0.00001419
0.00001660
0.00001866
  
```

Рисунок 5 – Результат выполнения программы linS.py

2	x	100	200	300	400	500	600	700	800	900	4500,000000000
3	y	0,0000023400	0,0000041700	0,0000062300	0,0000084900	0,0000110700	0,0000138700	0,0000164700	0,0000192800	0,0000212600	0,0001031800
4	x^2	10000	40000	90000	160000	250000	360000	490000	640000	810000	2850000,000000000
5	y*x	0,000234	0,000834	0,001869	0,003396	0,005535	0,008322	0,011529	0,015424	0,019134	0,0662770000
6	y^2	0,0000000000548	0,00000000001739	0,00000000003881	0,00000000007208	0,00000000012254	0,00000000019238	0,00000000027126	0,00000000037172	0,00000000045199	0,00000000154365
7	y	0,00000026	0,00000297	0,00000568	0,00000839	0,00001110	0,00001381	0,00001652	0,00001923	0,00002194	

Рисунок 6 – Таблица значений в excel

При помощи метода наименьших квадратов вывели систему уравнений: $2850000a + 4500b = 0,06627$ и $4500a + 9b = 0,000103$, решив которую мы нашли график функции: $y = 0,0000000271x - 0,0000024472$ и проверили при помощи метода корреляции правильность графика.

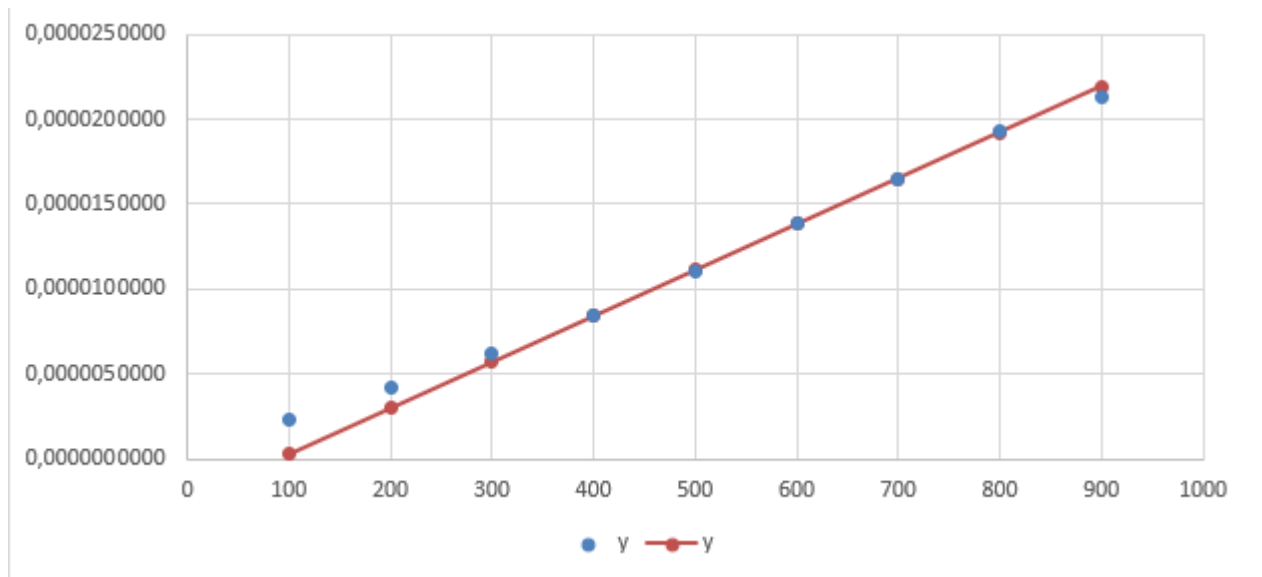


Рисунок 7 – График функции $y = 0,0000000271x - 0,0000024472$

11	г-исх	0,998291599
12	R-кв	1

Рисунок 8 – Проверка метода наименьших квадратов

Вывод: в результате проделанной работы стало понятно, что при наихудшем и при среднем случае существует зависимость выполнения алгоритма от количества элементов в массиве в котором происходит поиск.